

Impact of Lexical Filtering on Overall Opinion Polarity Identification

Franco Salvetti[†]
Stephen Lewis[#]
Christoph Reichenbach[†]

March 21, 2004

[†] Dept. of Computer Science, [#] Dept. of Linguistics, University of Colorado at Boulder

The Problem

- Opinion Polarity – thumbs up or down
- Supervised machine learning approach
- Impact of Lexical Filtering of the training set

The Data

- 27,000 movie reviews (Internet Movie Database)
- Extract 1500/1500 clearly positive/negative
- Test set of size 100/100 reviews
- Training set: 100–1500 reviews

Unigrams (Naïve Bayes)

- POS tagged data
- data containing only the adjectives
- data containing only the WordNet synsets of the adjectives

Results with Unigrams

| Training Size | All-POS | JJ | Synsets on JJ |
|---------------|---------|------|---------------|
| 100 | .615 | .640 | .650 |
| 400 | .740 | .700 | .730 |
| 700 | .775 | .715 | .710 |
| 1000 | .765 | .755 | .720 |
| 1500 | .795 | .730 | .735 |

Bigrams (Markov Model)

- POS tagged data
- Tagged data with POS filters
- Tagged data with WordNet filters

POS Filtering

| Rule | Example |
|-------------|-------------------|
| Keep | good/JJ → good/JJ |
| Discard | and/CC → |
| Generalize | movies/NNS → /NN |

Keep Discard Generalize

Example:

| | | | | | | | | |
|-----------|-----------|------------|------------|-----------|------------|-----------|-----------|------------|
| JJ | RB | VBG | VBN | NN | VBZ | CC | DT | COP |
| K | K | K | K | G | D | D | D | K |

“The/DT character/NN is/VBZ written/VBN thinly/RB”

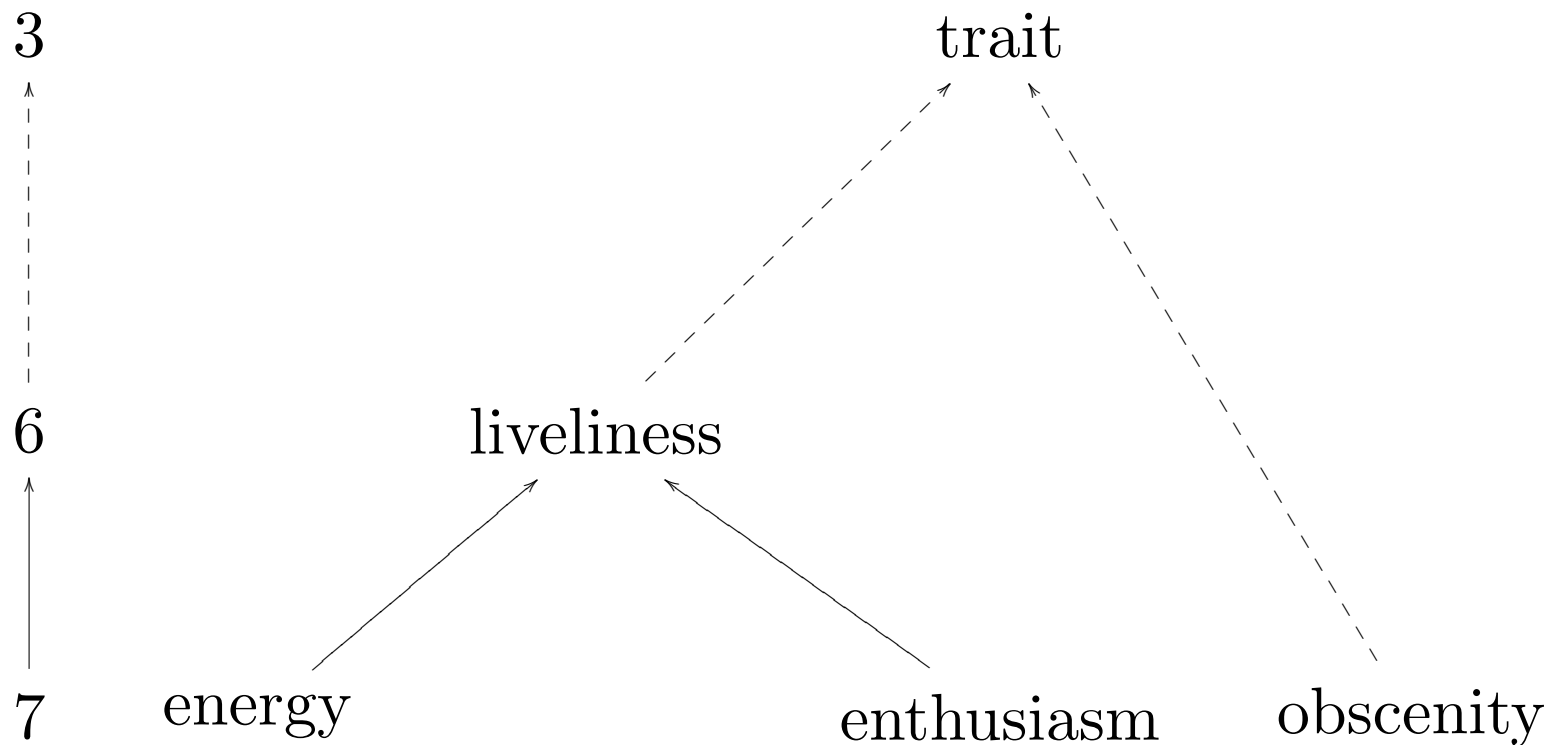
becomes

“/NN is/COP written/VBN thinly/RB”

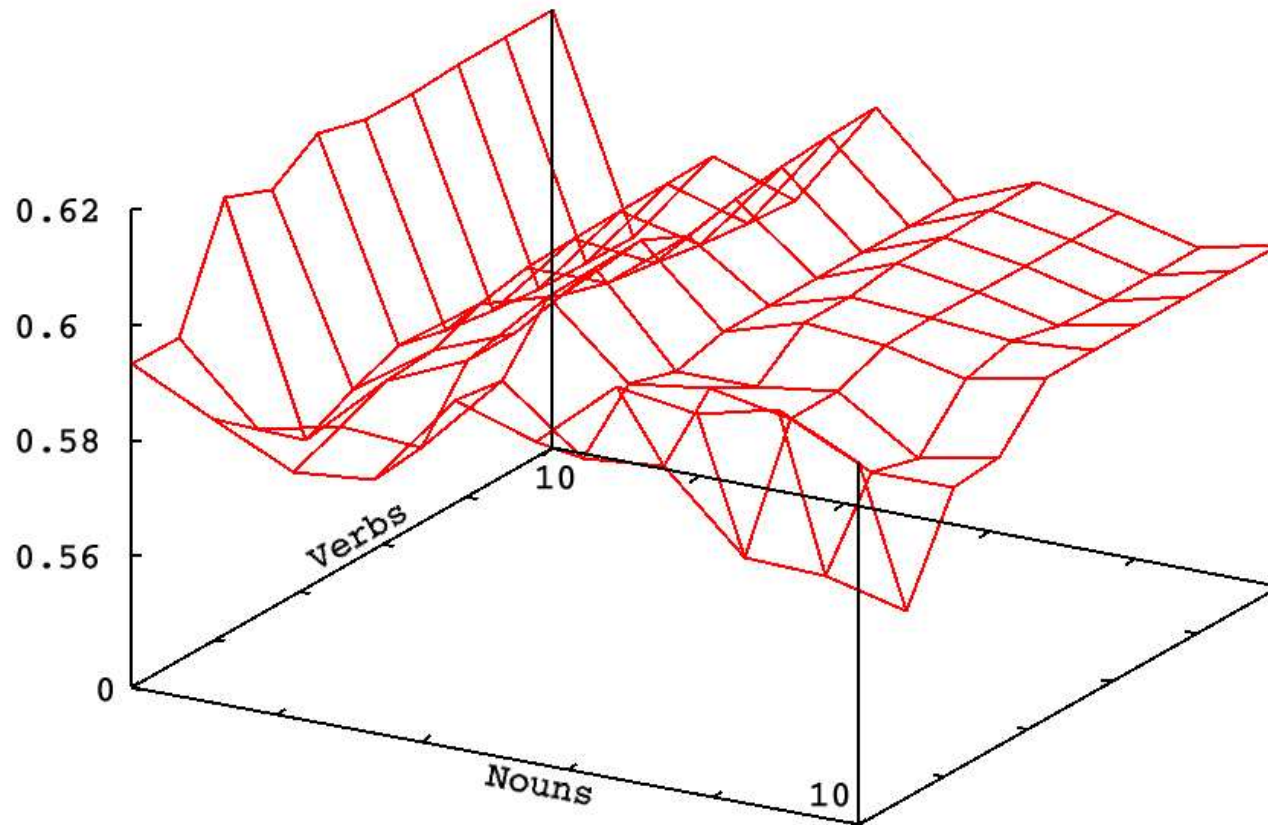
POS filtering with Bigram classifier

| Training Size | <i>Rule</i> ₁ | <i>Rule</i> ₂ | <i>Rule</i> ₃ | <i>Rule</i> ₄ | <i>Rule</i> ₅ | All-POS |
|---------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|---------|
| 100 | .555 | .625 | .625 | .630 | .630 | .575 |
| 400 | .700 | .660 | .660 | .685 | .685 | .710 |
| 700 | .705 | .700 | .700 | .690 | .690 | .735 |
| 1000 | .730 | .745 | .745 | .730 | .730 | .765 |
| 1500 | .725 | .730 | .730 | .750 | .750 | .770 |

Hypernym Generalization in WordNet



WordNet Generalization – small training set



Score by Odds Ratio

- $score = \log P(+|review) - \log P(-|review)$
- $P(+|review)$: Probability of review being positive
- $P(-|review)$: Probability of review being negative

Rank by Score

- The reviews are sorted by *score* (e.g. odds ratio)
- Label the first N reviews *positive* and the last N *negative*

| N | precision | recall |
|-----|-----------|--------|
| 10 | 1.000 | .100 |
| 30 | .900 | .270 |
| 70 | .830 | .580 |
| 90 | .780 | .680 |

Conclusions

- Filtering improves results on small training sets
- Semantic compression does not degrade accuracy
- Ranking gives 100% accuracy if we accept 10% recall

Questions. . .

. . . thank you.

Preprocessing Filtering

| Rule | Example |
|-------------|-----------------------|
| Copula | is/VB → is/COP |
| Copula | seems/VBZ → seems/COP |
| Negation | not/RB → /NEG |

Hypernymy in WordNet: 3000 reviews

